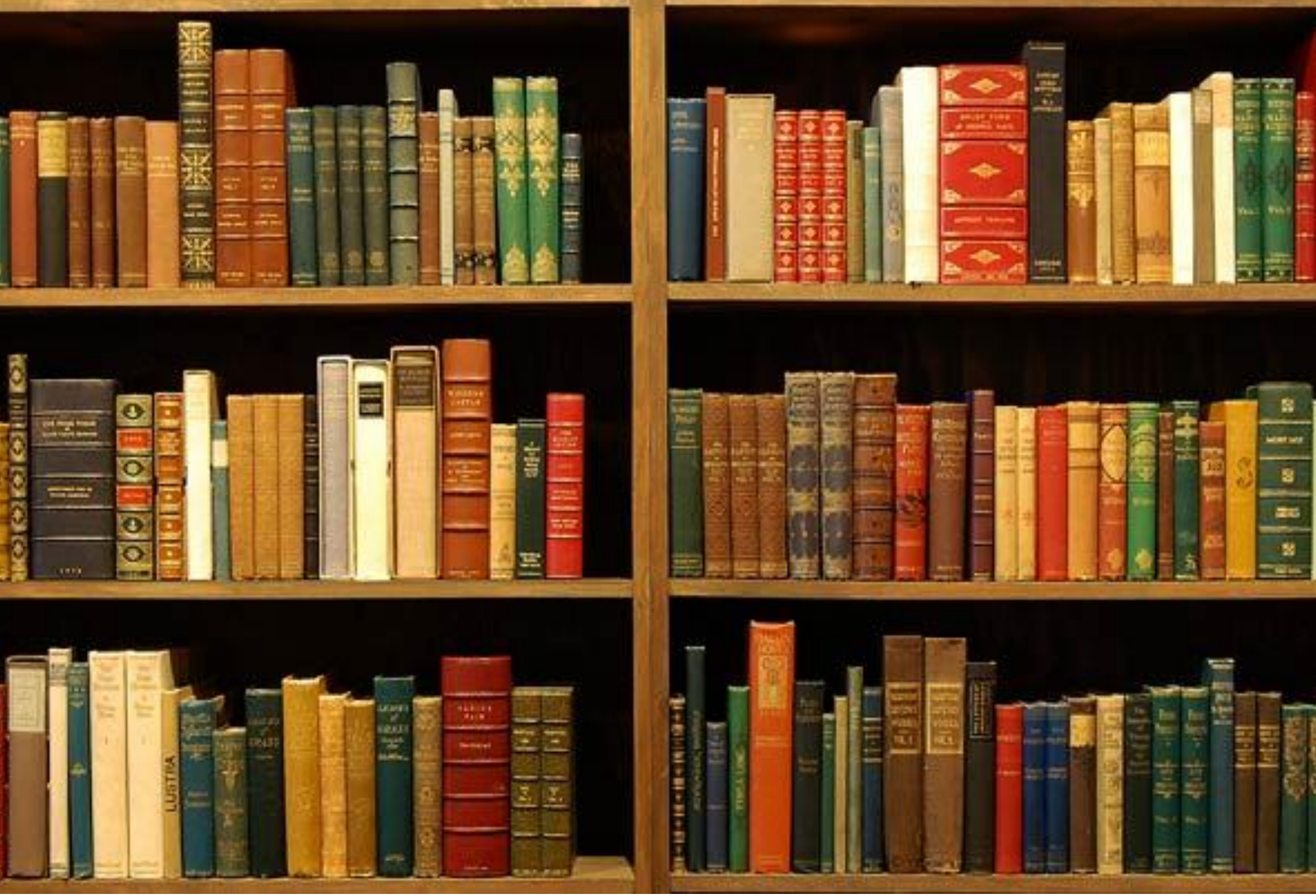
Statistical text mining using R

Tom Liptrot
The Christie Hospital



Example 1: Dickens to matrix

Example 2: Electronic patient records

Dickens to Matrix: a bag of words

IT WAS the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness, it was the spring of hope, it was the winter of despair, we had everything before us, we had nothing before us, we were all going direct to Heaven, we were all going direct the other way- in short, the period was so far like the present period, that some of its noisiest authorities insisted on its being received, for good or for evil, in the superlative degree of comparison only.

Dickens to Matrix: a matrix

Words

Documents

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

```
#Example matrix syntax
A = matrix(c(1, rep(0,6), 2), nrow = 4)
library(slam)
S = simple_triplet_matrix(c(1, 4), c(1, 2), c(1, 2))
library(Matrix)
M = sparseMatrix(i = c(1, 4), j = c(1, 2), x = c(1, 2))
```

Dickens to Matrix: tm package

```
library(tm) #load the tm package
corpus_1 <- Corpus(VectorSource(txt)) # creates a 'corpus' from a vector


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

it was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of light, it was the season of darkness, it was the spring of hope, it was the winter of despair, we had everything before us, we had nothing before us, we were all going direct to heaven, we were all going direct the other way- in short, the period was so far like the present period, that some of its noisiest authorities insisted on its being received, for good or for evil, in the superlative degree of comparison only.

Dickens to Matrix: stopwords

```
library(tm)
corpus_1 <- Corpus(VectorSource(txt))


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

it was the best **of** times, **it was the** worst of times, **it was the** age **of** wisdom, **it was the** age **of** foolishness, **it was the** epoch **of** belief, **it was the** epoch **of** incredulity, **it was the** season **of** light, **it was the** season **of** darkness, **it was the** spring of hope, **it was the** winter **of** despair, **we had** everything **before us**, **we had** nothing **before us**, **we were all** going direct **to** heaven, **we were all** going direct **the other** way- **in** short, **the** period **was so** far like **the** present period, **that some of its** noisiest authorities insisted **on its being** received, **for** good **or for** evil, **in the** superlative degree **of** comparison **only**.

Dickens to Matrix: stopwords

```
library(tm)
corpus_1 <- Corpus(VectorSource(txt))


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

best times, worst times, age wisdom, age foolishness, epoch belief, epoch incredulity, season light, season darkness, spring hope, winter despair, everything us, nothing us, going direct heaven, going direct way- short, period far like present period, noisiest authorities insisted received, good evil, superlative degree comparison .

Dickens to Matrix: punctuation

```
library(tm)
corpus_1 <- Corpus(VectorSource(txt))


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

best times worst times age wisdom age foolishness epoch belief epoch incredulity season light season darkness spring hope winter despair everything us nothing us going direct heaven going direct way short period far like present period noisiest authorities insisted received good evil superlative degree comparison

Dickens to Matrix: stemming

```
library(tm)
corpus_1 <- Corpus(VectorSource(txt))


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

best **time** worst **time** age wisdom age **foolish** epoch belief epoch **incredul** season light season **dark** spring hope winter despair **everyth** us **noth** us **go** direct heaven **go** direct way short period far like present period noisiest **author insist** **receiv** good evil **superl degre** comparison

Dickens to Matrix: cleanup

```
library(tm)
corpus_1 <- Corpus(VectorSource(txt))


corpus_1 <- tm_map(corpus_1, content_transformer(tolower))
corpus_1 <- tm_map(corpus_1, removeWords, stopwords("english"))
corpus_1 <- tm_map(corpus_1, removePunctuation)
corpus_1 <- tm_map(corpus_1, stemDocument)
corpus_1 <- tm_map(corpus_1, stripWhitespace)
```

best time worst time age wisdom age foolish epoch belief epoch incredul season light season dark spring hope winter despair everyth us noth us go direct heaven go direct way short period far like present period noisiest author insist receiv good evil superl degre comparison

Dickens to Matrix: Term Document Matrix

```
tdm <- TermDocumentMatrix(corpus_1)

<<TermDocumentMatrix (terms: 35, documents: 1)>>
Non-/sparse entries: 35/0
Sparsity           : 0%
Maximal term length: 10
Weighting          : term frequency (tf)

class(tdm)
[1] "TermDocumentMatrix"    "simple_triplet_matrix"

dim (tdm)
[1] 35  1
```

age	2	epoch	2	insist	1	short	1
author	1	everyth	1	light	1	spring	1
belief	1	evil	1	like	1	superl	1
best	1	far	1	noisiest	1	time	2
comparison	1	foolish	1	noth	1	way	1
dark	1	good	1	period	2	winter	1
degre	1	heaven	1	present	1	wisdom	1
despair	1	hope	1	receiv	1	worst	1
direct	2	incredul	1	season	2		

Dickens to Matrix: Ngrams

Google books Ngram Viewer

Graph these comma-separated phrases: best of times,worst of times case-insensitive

between 1800 and 2000 from the corpus English with smoothing of 3 . Search lots of books

Dickens to Matrix: Ngrams

```
Library(Rweka)
four_gram_tokeniser <- function(x, n) {
        RWeka:::NGramTokenizer(x, RWeka:::Weka_control(min = 1, max = 4))
        }

tdm_4gram <- TermDocumentMatrix(corpus_1,
                control = list(tokenize = four_gram_tokeniser)))

dim(tdm_4gram)
[1] 163   1
```

age	2	author insist receiv good	1	dark	1
age foolish	1	belief	1	dark spring	1
age foolish epoch	1	belief epoch	1	dark spring hope	1
age foolish epoch belief	1	belief epoch incredul	1	dark spring hope winter	1
age wisdom	1	belief epoch incredul season	1	degre	1
age wisdom age	1	best	1	degre comparison	1
age wisdom age foolish	1	best time	1	despair	1
author	1	best time worst	1	despair everyth	1
author insist	1	best time worst time	1	despair everyth us	1
author insist receiv	1	comparison	1	despair everyth us noth	1

Electronic patient records: Gathering structured medical data

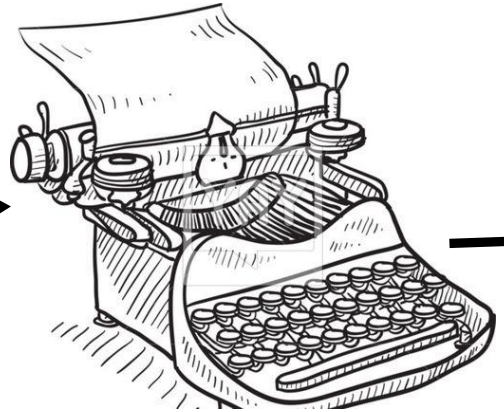
Doctor enters structured data directly

* Primary Disease Site	Oesophagus, Lower third
* Histology	Carcinoma, squamous cell
* Differentiation	Not known
* Clinical Stage	T3 N2 M0

* ECOG Performance Status	ECOG 1
* ACE Comorbidities Click to select the patient comorbidities	Overall Comorbidity Score is 2 = Moderate Venous Disease grade 2 Stomach / Intestine disorder grade 1

Electronic patient records: Gathering structured medical data

Doctor enters structured data directly



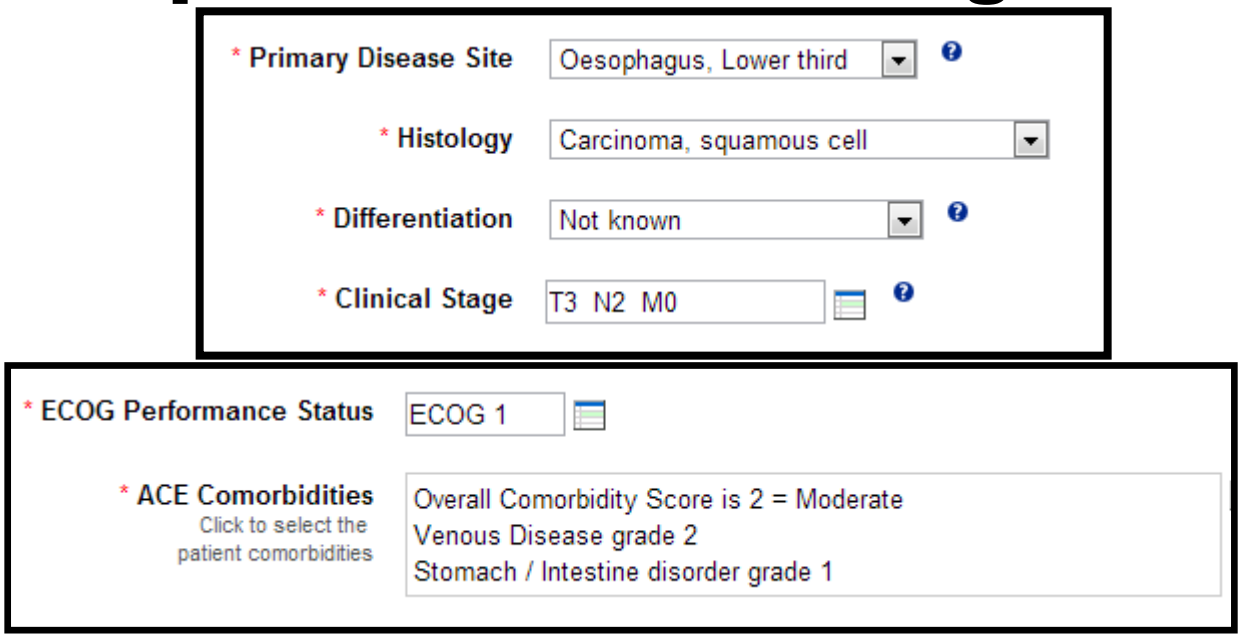
Trained staff extract structured data from typed notes

Electronic patient records: example text

Diagnosis: Oesophagus lower third squamous cell carcinoma, T3 N2 M0

History: X year old lady who presented with progressive dysphagia since X and was known at X Hospital. She underwent an endoscopy which found a tumour which was biopsied and is a squamous cell carcinoma. A staging CT scan picked up a left upper lobe nodule. She then went on to have an EUS at X this was performed by Dr X and showed an early T3 tumour at 35-40cm of 4 small 4-6mm para-oesophageal nodes, between 35-40cm. There was a further 7.8mm node in the AP window at 27cm, the carina was measured at 28cm and aortic arch at 24cm, the conclusion T3 N2 M0. A subsequent PET CT scan was arranged-see below. She can manage a soft diet such as Weetabix, soft toast, mashed potato and gets occasional food stuck. Has lost half a stone in weight and is supplementing with 3 Fresubin supplements per day. Performance score is 1.

Electronic patient records: targets



Diagnosis: **Oesophagus lower third squamous cell carcinoma, T3 N2 M0**

History: X year old lady who presented with progressive **dysphagia** since X and was known at X Hospital. She underwent an endoscopy which found a tumour which was biopsied and is a **squamous cell carcinoma**. A staging CT scan picked up a left upper lobe nodule. She then went on to have an EUS at X this was performed by Dr X and showed an early **T3** tumour at 35-40cm of 4 small 4-6mm **para-oesophageal nodes,** between 35-40cm. There was a further 7.8mm node in the AP window at 27cm, the carina was measured at 28cm and aortic arch at 24cm, the conclusion **T3 N2 M0**. A subsequent PET CT scan was arranged-see below. She can manage a soft diet such as Weetabix, soft toast, mashed potato and gets occasional food stuck. Has lost half a stone in weight and is supplementing with 3 Fresubin supplements per day. **Performance score is 1**.

Electronic patient records: steps

1. Identify patients where we have both structured data and notes (c.20k)
2. Extract notes and structured data from SQL database
3. Make term document matrix (as shown previously) (60m x 20k)
4. Split data into training and development set
5. Train classification model using training set
6. Assess performance and tune model using development set
7. Evaluate system performance on independent dataset
8. Use system to extract structured data where we have none

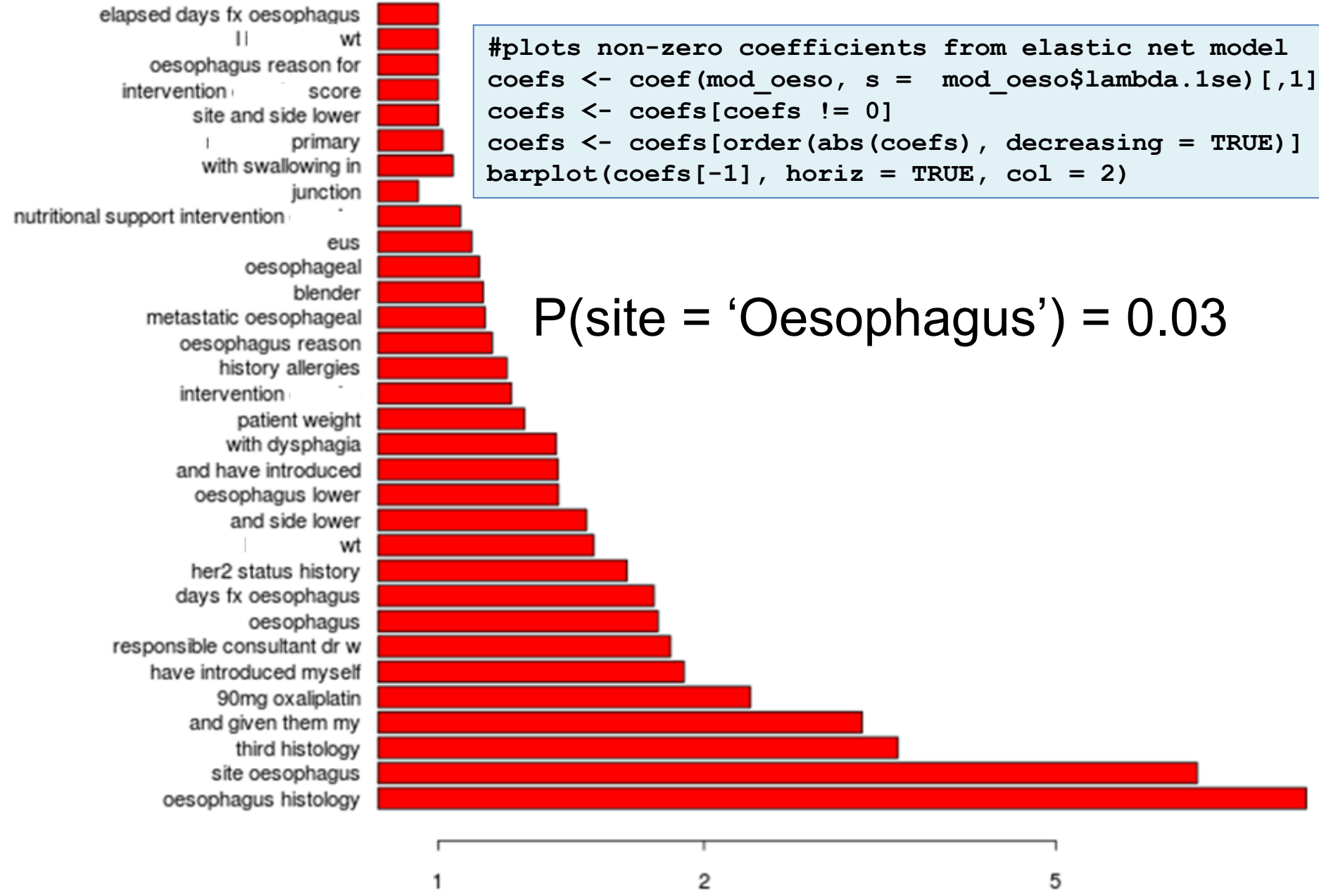
Electronic patient records: predicting disease site using the elastic net

OLS + RIDGE + LASSO

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}}(\|y - X\beta\|^2 + \lambda_2\|\beta\|^2 + \lambda_1\|\beta\|_1)$$

```
#fits a elastic net model, classifying into oesophagus or not
selecting lambda through cross validation

library(glmnet)
dim(tdm) #22,843 documents, 677,017 Ngrams
#note tdm must either be a matrix or a SparseMatrix NOT a
simple_triplet_matrix

mod_oeso <- cv.glmnet( x = tdm,
                       y = disease_site == 'Oesophagus',
                       family = "binomial")
```

Electronic patient records: The Elastic Net




```
#plots non-zero coefficients from elastic net model
coefs <- coef(mod_oeso, s =  mod_oeso$lambda.1se)[,1]
coefs <- coefs[coefs != 0]
coefs <- coefs[order(abs(coefs), decreasing = TRUE)]
barplot(coefs[-1], horiz = TRUE, col = 2)
```

P(site = ‘Oesophagus’) = 0.03

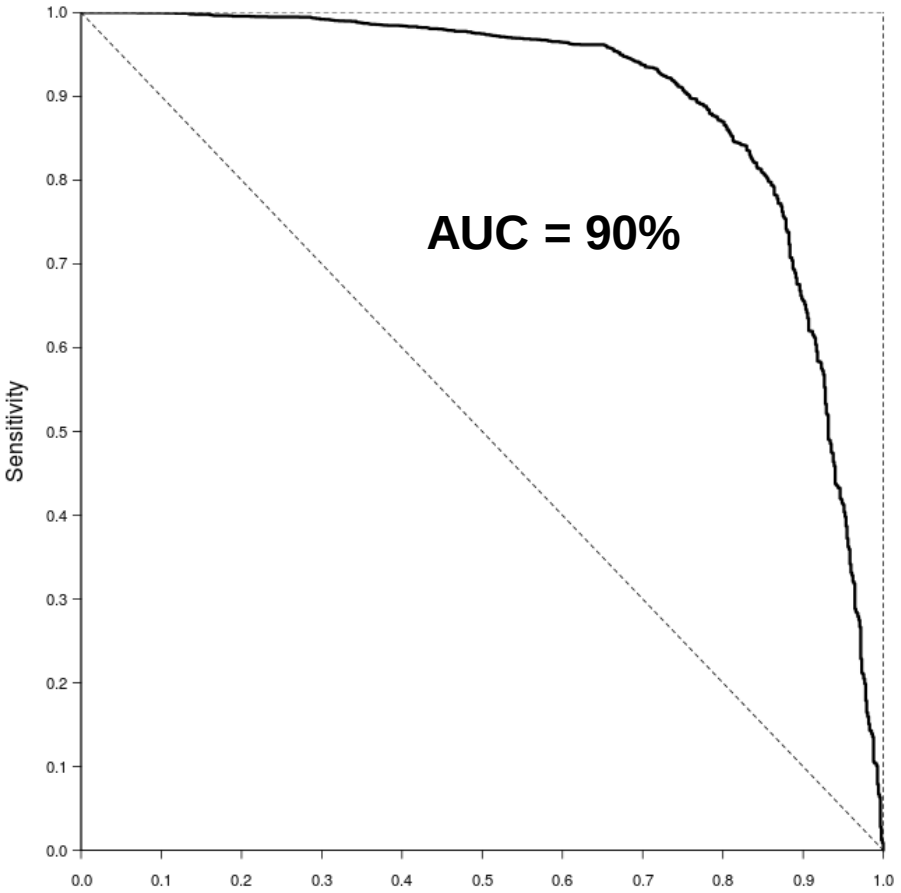
Electronic patient records: classification performance: primary disease site

Training set = 20,000
Test set = 4,000 patients

80% of patients can be classified with 95% accuracy (remaining 20% can be done by human abstractors)

Next step is full formal evaluation on independent dataset

Working in combination with rules based approach from Manchester University



Electronic patient records: Possible extensions

- Classification (hierarchical)
- Cluster analysis (KNN)
- Time
- Survival
- Drug toxicity
- Quality of life

Thanks

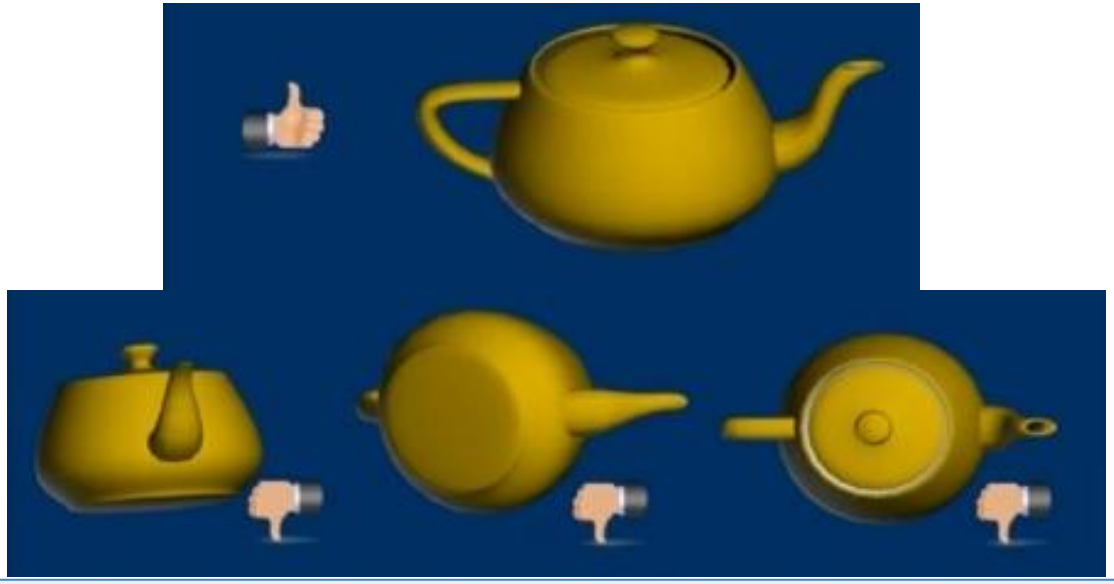
Tom.liptrot@christie.nhs.uk

Books example

```
get_links <- function(address, link_prefix = '', link_suffix = ''){
        page <- getURL(address)
        # Convert to R
        tree <- htmlParse(page)
        ## Get All link elements
        links <- xpathSApply(tree,  path = "//*/a",
                                    fun = xmlGetAttr, name = "href")
        ## Convert to vector
        links <- unlist(links)

        ## add prefix and suffix
        paste0(link_prefix, links, link_suffix)
        }


links_authors <- get_links("http://textfiles.com/etext/AUTHORS/", '/',
        link_prefix ='http://textfiles.com/etext/AUTHORS/')

links_text <- alply(links_authors, 1,function(.x){
        get_links(.x,    link_prefix =.x ,  link_suffix = '')
        })

books <- llply(links_text, function(.x){
        aaply(.x, 1, getURL)
        })
```

Text	Description
ARISTOTLE	Aristotle (384 BC - 322 BC)
BURROUGHS	Edgar Rice Burroughs (1875-1950)
DICKENS	Charles Dickens (1812-1870)
DOYLE	Sir Arthur Conan Doyle (1859-1930)
EMERSON	Ralph Waldo Emerson (1803-1882)
HAWTHORNE	Nathaniel Hawthorne (1804-1864)
IRVING	Washington Irving (1783-1859)
JEFFERSON	Thomas Jefferson (1743-1826)
KANT	Immanuel Kant (1724-1804)
KEATS	John Keats (1795-1821)
MILTON	John Milton (1608-1674)
PLATO	Plato (circa 428-c. 347 BC)
POE	Edgar Allan Poe (1809-1849)
SHAKESPEARE	William Shakespeare (1564-1616)
STEVENSON	Robert Louis Stevenson (1850-1894)
TWAIN	Mark Twain (Samuel Clemens) (1835-1910)
WILDE	Oscar Wilde (1854-1900)

Principle components analysis



```
## Code to get the first n principal components
## from a large sparse matrix term document matrix of class
dgCMatrix
library(irlba)

n       <- 5 # number of components to calculate
m       <- nrow(tdm) # 110703 terms in tdm matrix

xt.x    <- crossprod(tdm)
x.Means         <- colMeans(tdm)
xt.x    <- (xt.x - m * tcrossprod(x.means)) / (m-1)

svd     <- irlba(xt.x, nu=0, nv=n, tol=1e-10)
```

PCA plot

```
plot(svd$v[i,c(2,3)] + 1,
          col = books_df$author,
          log = 'xy',
         xlab = 'PC2',
         ylab = 'PC3')
```